Remarks

This response is to the Office Action mailed in the above-referenced case on January 04, 2005. Claims 1-14 are present below for examination. The Examiner has rejected claim 1 under 35 U.S.C. 102(e) as being anticipated by Theodore Carpenter-Smith et al. (hereinafter Carpenter-Smith). Claims 2-14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carpenter-Smith in view of Donald Baisley (U.S. 6,684,386) hereinafter Baisley, and further in view of "Metamodeling in OO" by Hafedh Mili et al., hereinafter Mili.

Applicant has carefully reviewed the Examiner's rejections and comments. Applicant herein provides argument to show where the art provided by the Examiner fails to provide a valid prima facie rejection.

Regarding claim 1, the Examiner states that Carpenter-Smith anticipates all of the limitations of the claim. The Examiner supports this rejection by creating a two column argument presenting applicant's claim in the left column and the portions of Carpenter-Smith, relied upon by the examiner, in the right column.

Applicant's claim 1 recites a limitation providing a drawing tool for visually representing *system views* using model elements represented by icons. The Examiner labels this limitation (a). The Examiner relies upon Carpenter Smith, col. 10, lines 35-39 to read on this limitation.

Applicant argues that there is no teaching in Carpenter-Smith, in particular in the portion relied upon by the Examiner, which teaches that model elements are represented by icons. Carpenter-Smith clearly teaches that a word document is imported into the system wherein the objects in the word document are tagged by highlighting them (Table1, Table2, col. 9). Applicant's preamble of claim 1 clearly recites that the system uses object models shaped to define the system views (p.15; Model Map Specification Language).

Further to the above, applicant's claim 1 comprises a limitation stating a first specification language for defining the various elements of a drawing, labeled (b) by the

Examiner; and a second specification language for defining mapping between drawing icons and objects stored in an object repository of the software system, labeled (c) by the Examiner. In applicant's invention the software control interface includes a specification language providing a complete semantic for defining the various symbols, connectors and other attributes of a visual diagram for creating and editing abstract models. Applicant's specification also teaches that a mapping language is required in order to map diagram elements of a particular diagram type to meta model elements or a map specification file. In a map specification, IconId refers to the IconId specified in the diagram definition file.

Applicant argues that in Carpenter-Smith a user completes many of the tasks that are performed in applicant's invention using one or the other of the first and second specification language. The Examiner refers to column 6, lines 64-67, which states: "When the classes, attributes, behaviors and collaborations are defined...". Applicant points out that Figure 5, columns 6, lines 46-63 describes the manual process. User interfaces are provided to perform each step 192-206. It is this manual operation that applicant's invention in part seeks to avoid.

In the background portion of applicant's invention it is explained that an example of a commercial repository system is Entity Relationship (ER) modeling. Typically, repository systems are packaged with a visual modeling tool that supports these standard methodologies either directly or in the form of a bridge to third party modeling tools (e.g.: Rational Rose). However, they do not provide any built-in support that enables end-users to easily specify visual diagrammatic notation for modeling abstractions introduced by the end-users. An end-user either has to be content with a generic, non-diagrammatic interface or has to build a graphical system from scratch if they want to provide visual modeling capability to abstractions they introduce. Applicant argues that the invention of Carpenter-Smith is similar to the above example wherein the user essentially builds the system from scratch.

So it appears that the Examiner has related the limitations of claim 1 to Carpenter-Smith by semantics, rather than by actual equivalent elements. Applicant believes that

claim 1 is clearly patentable over Carpenter-Smith as argued above. Claims 2-5 are then patentable on their own merits, or at least as depended from a patentable claim.

Regarding claims 2-14 the Examiner admits that Carpenter-Smith fails to teach meta-modeling and "tri-level structure". The Examiner relies upon Baisley to teach a repository containing groups of metadata called models and a special kind of model, called a meta model, describes the content of other models. The Examiner continues by stating that Baisley teaches that meta models typically describe classes, associations or references, and data types. The Examiner states that it would have been obvious to supplement Carpenter-smith's disclosure of transforming a system into a visual representation by using meta modeling taught by Baisley, for the purpose of sharing or re-using information about data, and using multiple level abstraction as taught by Mili for the purpose of representing application objects (Mili col. 2, end of first paragraph).

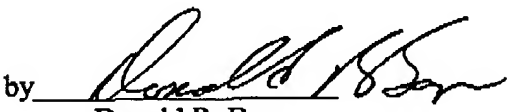
Applicant argues that the combination of the references as proposed by the Examiner cannot produce applicant's invention, firstly because Carpenter-Smith fails as a primary reference, and further because the Examiner has not shown where applicant's limitation of a user operating the drawing tool through a graphical user interface can, by dragging and dropping icons of the drawing tool onto a drawing sheet of the drawing tool, specify abstract models and model extensions expressed in diagrammatic notation transparent to the user and usable by the software system and associated repository.

The Examiner provides no references teaching providing software support for relating drawing tools in a software control interface to represent and specify abstract models and model extensions. The Examiner provides a drawing tool (Carpenter-Smith) and an example of a meta model (Baisley) and an example of multi-level abstraction (Mili), which together lack teaching of the claimed limitations providing the relations of the elements and functions in one software control interface for creating *and editing* system views. In applicant's invention, dragging and dropping icons of the drawing tool onto a drawing sheet of the drawing tool specifies abstract models and model extensions expressed in diagrammatic notation. Simply combining the references provided by the

Examiner, without using additional information from applicant's specification, cannot produce applicant's claimed invention. Therefore, the obviousness rejection fails as well.

Applicant believes claims 2-14 are therefore also patentable over the prior art, as argued above by applicant, and as argued on behalf of claim 1. Therefore, applicant respectfully requests reconsideration, and that the present case be passed quickly to issue. If there are any extensions of time required, such extensions are hereby requested. If there are any fees due, authorization is given to deduct the fees from deposit account 50-0534.

Respectfully Submitted,
Sreedhar Sannareddy Reddy

by 
Donald R. Boys
Reg. No. 35,074

Central Coast Patent Agency
P.O. Box 187
Aromas, CA 95004
(831) 726-1457